Introduction to Full Stack Development

1

LEARNING OBJECTIVES

After completing this chapter, you will be able to understand:

- Concept of full stack development.
- Essential technologies of full stack development.
- Web application development.
- Model–View–Controller with JSON, XML, etc.
- Object-relational mapping achieved through Hibernate.

- Front-end development with web technologies such as HTML, jQuery and Bootstrap.
- Working of Web services – API based architecture with REST.
- Back-end development with Java 11.

1.1 | Introduction

As the evolution of computer science escalated rapidly by the late 20th century, computers became smaller and more powerful. However, there was no central solution to digitally connect the masses. This was not possible until the invention of the Internet in the 1970s. Earlier, the Internet was not available for public use but by the start of the 1990s, businesses began to make inroads into the world of Internet connectivity. The advent of the Internet to the scene meant that physical shops were transformed into digital entities. Today, businesses and governments use the Internet to facilitate their operations.

The Internet mainly relies on the World Wide Web (WWW), also known simply as the Web, which uses HyperText Transfer Protocol (HTTP) to distribute information across various networks. This information is entailed in websites which are formatted in HyperText Markup Language (HTML). In the last three decades, the design, structure and capabilities of websites have changed a lot. From static websites to dynamic websites, and from dynamic to responsive websites catering to the demands of mobile users, there has been a great deal of change in the web sphere.

All these modifications were made possible due to the use of several web technologies. With time, many technologies such as Java's Applets and newer ones such as Node.js came to the scene. Today, full stack web development is the contemporary practice to develop websites.

In this chapter, you will learn several concepts that will give you an idea about the full stack development.

Fact Alert

At the time of writing this book, there are around 710 coding languages in the world.

1.2 | What is Full Stack Web Development?

In order to understand the full stack web development better, we can study a real-life example of an e-commerce application. Let us explore Amazon.com, a popular e-commerce platform known across the world for its innovative delivery mechanism. Amazon.com is widely accessible via a website www.amazon.com or via a mobile application. Both these platforms are known as *front-end* or *client-side*. These front-ends connect with other application that resides somewhere in a remote cloud server. This application is called *back-end application*. It provides services that can feed data to the front-ends. Front-ends can be developed in HTML, CSS, jQuery, Bootstrap, etc., and back-end can be developed in Java, C#, Python, Ruby on Rails, Node. js, etc. Since amazon.com is a private company and its platform is a proprietary one, it is difficult to predict the technologies they use.

1.2.1 Front-end

Usually, to design the view (or the client-side) of a web page which is visited by users through web browsers, we have the front-end development where HTML, CSS, and JavaScript are the fundamental technologies. All the menus, sliders, labels or anything you click or read on a website are generated with the help of front-end. It is all about graphics, how everything appears to users. Hence, front-end web development is also called *web design*.

1.2.2 Back-end

There is also a back-end, which is also known as the server-side. The back-end is used for business logic. It consists of a server which receives requests, an application which waits for requests and generates a response, and a database which stores all the data.

Traditionally, front-end and back-end are handled by separate professionals who have mastered any of these fields to power the website.

Full stack web development is a practice in which both the front-end and back-end are managed by the same professional. A full stack developer is not a master of a single domain; they have the conceptual knowledge and the technical expertise to create both the front-end and the back-end of websites from scratch. Such developers generally have experience in both domains, enabling them to gain an all-round understanding about all layers of web development.

The back-end contains two parts: (a) application layer and (b) persistence layer, also known as database layer. Figure 1.1 shows the three-tier architecture in which the front-end is referred as *client tier* and the back-end is divided into *application tier* and *database tier*. The application tier contains business logic and the database tier stores data generated by the application or entered by a user from the front-end tier.
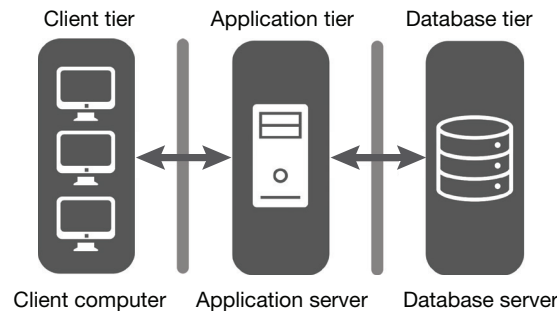

Figure 1.1 Three-tier architecture.

1.2.3 Technologies Essential for Full Stack Development

In order to become a sought-after developer, mastering the relevant technologies is essential. As we have learnt earlier, to work as a full stack developer, you need to learn technologies for all the layers shown in Figure 1.2.

Figure 1.2 Technologies for full stack application development.

1.3 | Introduction to Web Application Development

A web application exists on a server and is used when it is opened via web browsers. Web application development is the production of application programs that are run on servers and provide useful functionalities to the end-user with the help of the Internet. They include both client-side and server-side programming. They provide capabilities such as linking with databases, giving back a response to browsers, and performing a task for a user. E-commerce, social media websites, online banking are some examples of web application development. HTML, CSS, and JavaScript are some commonly used technologies for the front-end of a web application development. As shown in Figure 1.3, these technologies fall into the following categories:

1. Behavioral, which deals with behavior of an application such as actions performed on various events
2. Structural, which deals with forming the structure of an application such as adding tables, titles, etc.
3. Presentational, which focuses on the look and feel of an application like color, font style, alignment, etc.

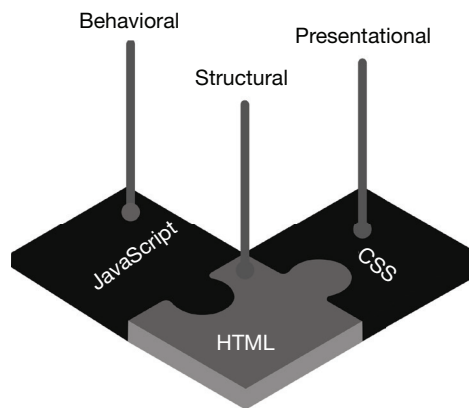**Figure 1.3**   Front-end technologies for web development.

> **Fact Alert**
>
> In 1999, Dale Dougherty initially coined the term "web development". Later it was popularized by Tim O'Reilly and Dale Dougherty in late 2004.

1.4 | Front-End Technologies

A tool to run front-end comprises numerous things – it can be a computer, laptop, mobile phone, tablet, smartwatch, car dashboard screen, etc. Although the development of each of these front-end types would be different, the basic concept of design decisions remains the same. You need to understand your end-user's expectations in order to design applications. The main aim of the front-end is to provide a user-friendly interface to allow interaction with backend and database. As a full stack developer, you must understand human–computer interaction (HCI). This field of study focuses on all aspects that ensure that the system is usable. In other words, HCI focuses on improving the design of computer systems which makes it easier for users to interact with them. However, since HCI is a vast area of study, we will not cover this topic entirely in this book. Figure 1.4 helps us to understand the elements that are involved in HCI.
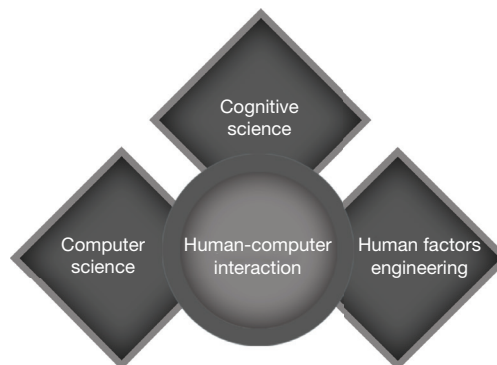
Figure 1.4   The multidisciplinary field of HCI.

In this book, we will study web-based application development. Hence, our front-end development focus will be on HTML, CSS, jQuery, and Bootstrap. Now, let us move on to understand these technologies a little bit. We will be covering each technology in detail in its own chapter, so for now, let us just introduce these briefly.

1.4.1 HTML/CSS

You must have a proficiency in HTML, which is an irreplaceable component of web development. HTML creates the arrangement of components of a web page with the help of markup. Markup powers HTML elements to use text and present it in a certain way. For instance, it can make some information on the page to appear "bold" on a website. The World Wide Web Consortium (W3C) is an organization that carries the responsibility to update it according to the ever-changing web scene.

As an example, let us see the following code which provides a title, heading, and a paragraph. While we have used Sublime Text 3 to write, one of the best things about HTML is that it does not require separate software. For learning purposes, you can even use notepad. However, in real-world development, web developers generally use an editor as it comes with lots of useful functionalities and plugins, which saves time and improve productivity.

```
<!DOCTYPE html> html
<html>
    <head>
        <title>What Is Full Stack Web Development?</title>
    </head>
    <body>
        <h1> My Heading: Full Stack Development</h1>
        <p>In this book we will learn about Full Stack Development.</p>
    </body>
</html>
```

You can use any text editor like notepad to paste the above code and save this file as "whatisfullstack.html". In this example, we are saving this file with ".html" extension. This extension is used to save an HTML file, which can be opened in any web browser. All browsers know how to render an HTML file. This way any browser can render this code and produce a result. If you run this file in any browser, you would get the following response.

However, HTML elements are not too visually appealing. This is where Cascading Style Sheets (CSS) come into the picture. As the name suggests, CSS adds "styles" to HTML components. Therefore, CSS is heavily used to provide striking visuals to the web pages for incorporating layouts, designs, and variations.

While HTML did have its own features to add style to web pages, it became too complex and unmanageable to add colors or fonts to each web page separately. Hence, CSS gained recognition, where all the styling-related work was administered through separate style sheets. As an example, let us add red color to our previous HTML example and also put it across the center.

```
<!DOCTYPE html>
<html>
    <style>
        h1 {
         color: red;
         text-align: center;
         }
    </style>
    <head>
        <title>What Is Full Stack Web Development?</title>
    </head>
    <body>
        <h1> My Heading : Full Stack Development</h1>
        <p>In this book we will learn about Full Stack Development.</p>
    </body>
</html>
```

You can either add this code in the same file or create a new one. For keeping these two examples separate, we have created two different files. You can just add style code above <head> tag and save.

```
<style>
        h1 {
         color: red;
         text-align: center;
         }
</style>
```

Upon opening this file in a browser, you will see the following image

Fact Alert

Sir Tim Berners-Lee, a physicist at CERN and the inventor of the Internet, first proposed and prototyped ENQUIRE, a system for CERN researchers to share documents. Later in the year of 1989, he proposed an Internet-based hypertext system and in the subsequent year (1990), defined HTML and developed browser and server.

QUICK CHALLENGE Add more elements like labels, buttons, radio buttons, etc. and render it in a browser. Make sure the header color is blue and the text is aligned right.
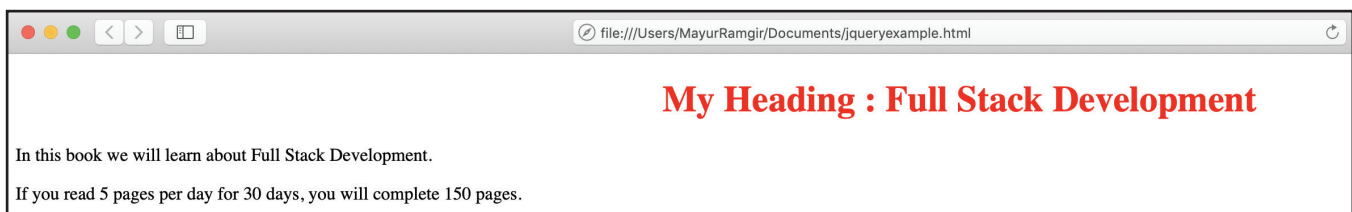
1.4.2 JavaScript

After HTML and CSS, we have JavaScript – the most in-demand language of recent years in the technology community. Unlike the former two front-end technologies, JavaScript is a proper high-level programming language, a scripting one. JavaScript enables developers to add more sophisticated functionality on the client-side. It transforms a web page from a static to a dynamic one by adding interactivity and timely features such as animations, advanced maps, etc. All the common web browsers have a dedicated engine to process the language on the client-side.

Today, JavaScript has popular front-end frameworks like Angular JS, Vue JS, React JS, etc. While it was originally used for the client-side, it has come on the back-end with Node.js. For a simple JavaScript example, let us see the following. We will use the same file to add a simple JavaScript code to multiply two elements.

```html
<!DOCTYPE html>
<html>
    <head>
        <title>What Is Full-Stack Web Development?</title>
        <script>
                var pagesperday = 5;
                var readingdays = 30;
                var daysneeded = pagesperday * readingdays;
                document.getElementById("result").innerHTML = daysneeded;
        </script>
        <style>
                h1 {
                    color: red;
                    text-align: center;
                }
        </style>
    </head>
    <body>
        <h1> My Heading : Full Stack Development</h1>
        <p>In this book we will learn about Full Stack Development.</p>
        <p>
            If you read 5 pages per day for 30 days, you will complete <span
                id="result"></span> pages.
        </p>
    </body>
</html>
```

This code will give us the following result. In this, we are doing a simple multiplication of two variables and placing the result in an HTML file which has id as "result".

QUICK CHALLENGE Write codes to perform various other functions like addition, subtractions, etc.

For this book, we will focus on jQuery as a JavaScript library.

1.4.3 jQuery

jQuery is a small but powerful JavaScript library. It is used to ease scripting on the client-side with HTML. It can be used for traversing HTML documents or modifying them. Likewise, it is used for event handling and producing animations. The primary objective of jQuery is to add interactivity on a website. Since it can help to write shorter codes than JavaScript, therefore it is also called "write less do more". jQuery is a cross-platform. For a simple jQuery example, let us see the following. This example adds dynamicity to HTML as it hides the heading by a simple button click.

```html
<!DOCTYPE html>
<html>
    <head>
        <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js">
</script>
        <script>
            $(document).ready(function() {
                $("button").click(function() {
                    $("h2").hide();
                });
            });
        </script>
    </head>
    <body>
        <h2>This element will get removed once the following button is clicked.</h2>
        <p>JQuery Example Code</p>
        <button>Hide Heading</button>
    </body>
</html>
```

Save this code in the same way you saved the HTML code earlier and run in a browser window. Upon running this, you will see the following output.

file:///Users/MayurRamgir/Documents/jqueryexample2.html

This element will get removed once the following button is clicked.

JQuery Example Code

Hide Heading

QUICK CHALLENGE Take the above example and add code to replace text from the <p> tag. For example, change "JQuery Example Code" to "jQuery Content Replacement Example" on the button click.

Also, in order to run this application on mobile and tablet browsers, we need to use a library known as Bootstrap.

1.4.4 Bootstrap

As Nokia was dethroned by the likes of Apple and Samsung, several smartphones and tablets were introduced in the market in the late 2000s. At that time, websites were primarily designed for desktops and laptops. As more and more people used smart devices to surf the Internet, they found the web experience to be extremely unsatisfactory. To solve this issue, Twitter's Jacob Thornton and Mark Otto developed Bootstrap in 2011; hence, you might have often read it as "Twitter Bootstrap". Bootstrap was not a new front-end markup language, but it used the existing front-end technologies HTML, CSS, and JavaScript and paved the way for responsiveness web design.

Bootstrap is the most famous HTML, CSS, and JS framework for designing responsive, mobile-oriented projects on the web. In other words, Bootstrap is a huge collection of useful, reusable bits of code written in CSS, HTML, and JavaScript. It has become an important tool for front-end developers. Developers and designers can quickly create fully responsive websites with the help of Bootstrap.

Responsiveness is a metric which refers to the tailored rendering of a website with respect to its device and screen size. Hence, the advent of Bootstrap made it easier for web designers to make websites for a wider audience. However, responsiveness is not the only advantage brought to the table by Bootstrap. It also provides developers with templates for HTML and CSS to add tables, image carousels, modals, buttons, forms, typography, etc. on websites.

1.4.4.1 What are the Advantages of Using Bootstrap?

Using Bootstrap saves us from writing tons of CSS codes. Moreover, the best thing about Bootstrap is that it is free. Presently, Bootstrap is hosted on GitHub. Following are some of the other advantages of Bootstrap:

1. **Responsive grid:** There's no need to spend hours to code your own grid. Bootstrap provides you with its own grid system. By using it, developers can directly fill their containers with content. Setting up your custom breakpoints for each column is super easy by using their small, medium, and large breaks. You can just choose the default as it could already meet your requirements.
2. **Responsive images:** Bootstrap helps you to automatically resize and optimize according to the screen size by using its own code. Furthermore, you can also change the shape of the images. This could be easily done without continuously switching between the code and your design software.
3. **Components:** Bootstrap provides a handful of components which could be tracked onto your web page. This includes:
   - Navigation bars.
   - Dropdowns.
   - Progress bars.
   - Thumbnails.

   Not only will you be able to add eye-striking elements to your web page, but you will also be able to optimize them automatically according to different screen sizes. You do not need to do tons of work as there are a lot of ready-made functionalities are ready to use.
4. **JavaScript in Bootstrap:** Bootstraps provides developers with tons of jQuery plugins. jQuery facilitates with a greater level of interactivity. This creates easy solutions for modal popups, image carousels, transitions, etc.
5. **Documentation:** The document of Bootstrap is one of the best documentations in the software industry. Each piece of code is well described to the smallest detail in their official websites. These explanations also include code samples making it perfect for the beginners. You just need to copy a code and paste it into your page.
6. **Bootstrap community:** Just like any other open-source software, Bootstrap also has a very large and friendly community of developers and designers. Developers find it easy to modify and contribute to the Bootstrap's database as it is hosted on GitHub. People often collaborate, give useful pieces of advice, and interact with each other to solve doubts.

For a simple example, let us create the following table in Bootstrap.

```
<!DOCTYPE html>
<html lang="en">
    <head>
        <title>Bootstrap Example</title>
        <meta charset="utf-8">
        <meta name="viewport" content="width=device-width, initial-scale=1">
        <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css
/bootstrap.min.css">
        <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js">
</script>
        <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.
js"></script>
    </head>
    <body>
        <div class="container">
            <h2>Table Example</h2>
            <p></p>
            <table class="table">
                <thead>
                    <tr>
                        <th>Student Name</th>
                        <th>Course Marks</th>
                        <th>Total Marks</th>
                    </tr>
                </thead>
                <tbody>
                    <tr>
                        <td>Merry</td>
                        <td>Introduction to Computer Vision</td>
                        <td>95</td>
                    </tr>
                    <tr>
                        <td>Jim</td>
                        <td>Distributed Computing</td>
                        <td>63</td>
                    </tr>
                    <tr>
                        <td>Morrison</td>
                        <td>System and Software Security</td>
                        <td>74</td>
                    </tr>
                </tbody>
            </table>
        </div>
    </body>
</html>
```
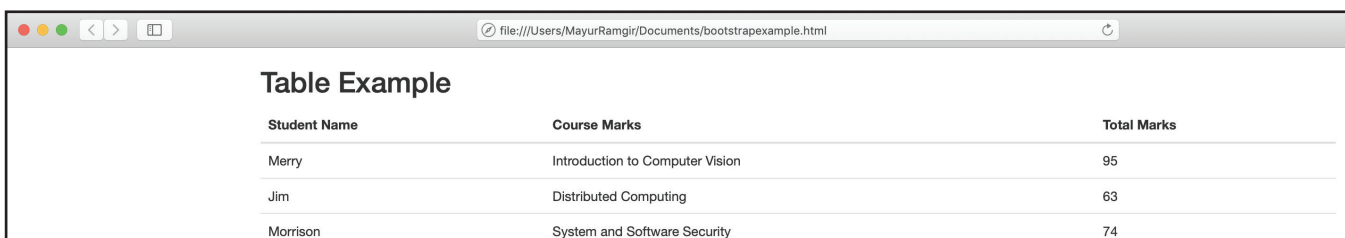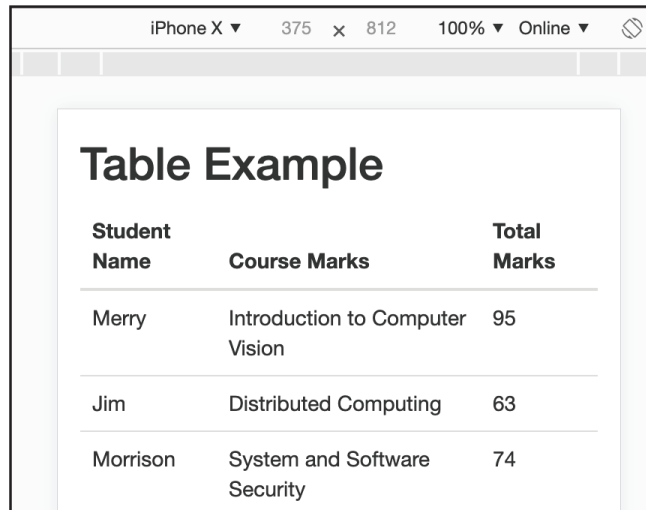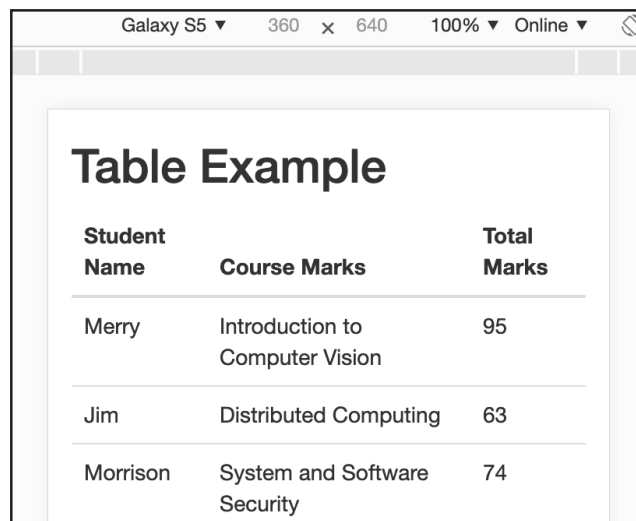
This program will produce the following result upon running in a browser window.

| | | file:///Users/MayurRamgir/Documents/bootstrapexample.html | | |

Table Example

Student Name	Course Marks	Total Marks
Merry	Introduction to Computer Vision	95
Jim	Distributed Computing	63
Morrison	System and Software Security	74

Using a simulator add-on for browsers, we can see how this code renders on different devices. Following are views on some devices.

iPhone X View
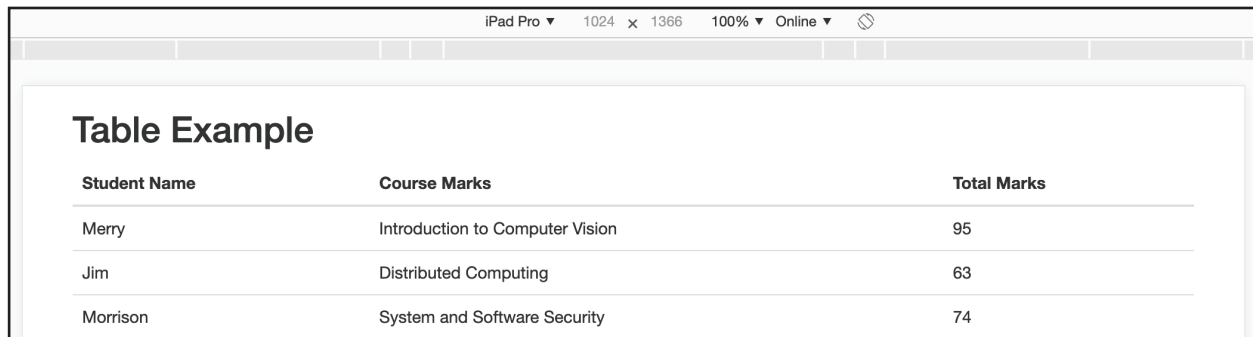
Samsung Galaxy S5 View

iPad Pro View

> **QUICK CHALLENGE**
> Create a new HTML file and write code to create four div elements. Use CSS to show them in four columns. Add an image element in each div. Use Bootstrap to make all four divs render properly on devices like iPhone, Samsung Galaxy, and iPad.

1.5 | Back-End Technologies (Server-Side)

The server-side requires the use of a single or multiple programming languages to write business logic, user authentication, or database (DB) related work. This is a matter of preference. Each back-end language has its pros and cons. There is no back-end language that is the best solution for all needs; each back-end language is good to solve a certain problem. As a computer programmer, you must be able to adapt because computer science always meant to be for "solving problems" rather than "solving problems with a specific language". We have the following options as back-end languages:

1. **PHP:** Easily the most common tool for designing the back-ends of a website. PHP is good for those who want convenience at the expense of functionality. Setting up websites with PHP requires minimal effort and time. It is generally used with frameworks such as Laravel, CodeIgniter, etc. However, there are various content management platforms such as WordPress, Joomla, etc. that are popular to run websites. These frameworks are customizable with plugins.

2. **Node JS:** This allows developers to write both the front-end and back-end in a single language – JavaScript. This is a great option for those who are proficient in the foundations of JavaScript, though there are still questions about Node JS maturity and capability to work in enterprise-level applications. Node JS is often used in a technology stack known as MEAN (short for MongoDB, Express, Angular, Node JS).

3. **Python:** This is increasingly becoming an all-purpose language. It is used heavily in Artificial Intelligence and Data Science for making computers "intelligent". Some also use it for desktop development as well as design management systems. Similarly, the web sphere has also accepted it. Python's charm comes with its ability to allow users to write only a few lines of code. While a developer may have to write a simple "Hello World" program in other languages in multiple lines, Python can execute it in a single line. Python is generally used in Web with Django and Flask frameworks.

4. **C#:** This is the brainchild of Microsoft, one of the giants ruling over the IT industry. C#, along with Java, is one of the two languages that is primarily used for coding enterprise-level development. C# itself is inspired from Java. Due to Microsoft's R&D, the language is used heavily across different businesses and industries where it is the core programming medium of "Microsoft" stack, which runs Microsoft OS like Windows Server, databases like SQL Server, and C# to design desktop application with Visual Studio or writing web applications with the ASP.NET framework.

5. **Java:** Java is perhaps the most used language in the world. It is used all around the world for banking ecosystems because of its high security. Universities use it to teach CS courses such as Object-Oriented Programming, Data Structures, etc. The world's leading mobile operating system Android depends on mobile apps which are written in Java. On the web, Java uses a multitude of frameworks such as Java EE, Play Framework, Vaadin, Struts, etc. However, the most famous one is Spring Model View Controller (MVC).

> **Fact Alert**
> Java programming language was originally named Oak. However, Oak was already taken by another company, which made the Java team to come up with a new name.

For this book, we will select Java as the back-end development language. Java has many advantages over other programming languages and has received huge support from the business world. It is the by default choice for many large-scale applications and has improved a lot over the years to keep its leading position. Let us study the latest version of Java, most of which we will learn in detail in later chapters.

1.6 | Introduction to Back-end Development with Java 11

There is a reason Java has cemented such a strong position in the global community over the past few decades. Its cross-platform computability with Java virtual machine (JVM) was a breakthrough that gave birth to the likes of Scala today. Some credit its success to its massive community while many believe that Java has a library for anything. Experienced programmers opine that it is actually the scalability and security factors that have made the language invaluable. Let us take for example Twitter, which

was designed using Ruby on Rails but broke at a point and had to adopt JVM for avoiding the crisis again. There are similar examples available throughout the industry where Java proved to be a lifesaver in adversity.

However, Java is not without its detractors. Earlier Java EE proved to be messy and complex. It was not the apparent vulnerability of the language that proved to be the negative point, but it was actually its difficulty level to work with that discouraged others. With Java EE (then J2EE), the first issue was working with a component which needed the configuration of XML files. The second issue was setting up the component dependency (e.g. whenever a component, say X, had to use a different component, say Y, then component X had to search by itself for component Y). Lastly, some services were not required by the components. Hence, there was the problem of "heavy weight".

Fortunately, with the release of modern-day Java frameworks such as Spring, Spring Boot, Play, etc., these issues have been addressed well.

Today, Spring is considered by many to be the leading Java framework. It is a lightweight framework which can be used for a plethora of web projects. Spring is known for working with a concept called *dependency injection.* Similarly, aspect-oriented programming is another fundamental concept of the framework. These concepts have been covered in detail in Chapter 20.

Spring has proved to be effective in removing boilerplate code and has been praised for its reduction of complexities, which proved to be a nightmare for developers working with Java EE.

Spring can be seen as a framework which provides integration of all Java API and technologies and enables their usage with plain old Java objects (POJOs). Hence, it is important to realize that Spring does not completely invent new technologies in Java's web framework. Instead, it gives a powerful and optimized solution through which technologies such as EJB, JMS, and Hibernate can be used easily.

Due to its wide support to integrate a long list of technologies, Spring framework is used for the creation of versatile, testable, and efficient code which is good for both traditional ecosystems and newer ones with Android or functional paradigms. Now, let us explore the MVC pattern and see how Spring MVC will be useful for the development.

1.7 | Introduction to Model View Controller (MVC)

Model View Controller (MVC) is an architectural pattern that is used to build software applications. Due to its effectiveness, the pattern has received huge adoption in the web space where you can see Spring MVC and ASP.NET MVC as one of the most well-known proponents of it. The pattern works by dividing an application into three layers, which help improve modularity and reusability of the application while they are more flexible and can support iterations.

To explain MVC, let us consider a simple example. You go to a coffee shop for your morning breakfast. You interact with a barista to order an Expresso. The barista communicates your order to others where a manager makes sure that your order is initiated and processed properly. The coffee machine and the staff use your information and make the right coffee. Afterward, the barista gives you the coffee. Here, the manager who masterminded everything is the Controller. The coffee machine and the staff who had the information can be referred to as the Model, while the barista who was on the "front" can be called as the View. Now going by this let us explain what each layer does.

1. **Model:** It carries out all the management of all the data-related logic. Likewise, the DB operations like SELECT, UPDATE, are performed at this layer. Mostly, it engages in communication with the controller while at times it may directly provide some data to the view.
2. **View:** It is the user interface (UI) which is seen by the users. All the front-end codes like HTML, CSS, and JavaScript are part of the view. The controller can forward dynamic values to the view for real-time updates.
3. **Controller:** It gets user-input from either a URL request or a view. It then goes on to perform actions on the requests. The view receives data from the controller while the model provides that data to the controller.

> **Flash ? Quiz** What is the importance of Model in the MVC framework?

1.8 | Introduction to Web Services: API-Based Architecture with REST

As each programming language possesses its own strong and weak points, a business may use PHP to build one web application while Java may prove to be more practical in another scenario. At times, there is a need to establish communication between these web applications that exist on separate technology stacks. This is where web services come into play. A web service is an

intermediary that is used for exchanging messages between the server and the client operating on the Web. Web services are programmed to execute multiple actions.

1.8.1 REST

REST is a type of web service; it competes with another type of web service known as Simple Object Access Protocol (SOAP). Some analysts find SOAP to be more powerful while some credit REST as it needs a lesser amount of bandwidth, thereby making it extremely suitable for the Internet.

Application Programming Interface (API) refers to lines of code that facilitate two different programs to engage in communication. API carves out a methodology for the coder through whom a program may go on to ask for services from any application or OS.

RESTful APIs work by decomposing a transaction into several little modules. Each of these modules is made to solve a different portion of the transaction. This may be a bit complex for developers to code but it reaps rewards with an enhanced degree of modularity. RESTful APIs use HTTP protocol and its standards. Such an API may use GET for the retrieval of a resource. Similarly, it can use PUT for making an update or change in a resource. Likewise, it may generate a resource by POST or eliminate it by utilizing DELETE.

In REST architecture, calls do not carry any state. Thus, they are a good fit for cloud-based environments. Components that are stateless are advantageous because they can be easily redeployed at any failure while they can also support scaling for load changes. The reason for this is that requests can be easily sent to any of the component's instance.

> **Flash ? Quiz** Why is modularity important in developing a scalable application?

1.8.2 Simple Object Access Protocol (SOAP)

Simple Object Access Protocol (SOAP) is a messaging protocol that makes it possible for the distributed elements of an application to communicate with each other. SOAP uses eXtensible Markup Language (XML) to enable communication among web services. Its main aim is to facilitate neutrality, extensibility, and independence. It can also be used for broadcasting a message.

SOAP facilitates processes operating on disparate operating systems (e.g. Windows and Linux) to contact using XML.

1.8.2.1 Characteristics of SOAP

Following are the main characteristics of SOAP:

1. **Neutrality:** SOAP can function over any protocol such as HTTP, SMTP, TCP, UDP, and JMS.
2. **Extensibility:** Security and WS-Addressing are one of the top priorities of SOAP.
3. **Independence:** SOAP allows any kind of programming model.

Even though SOAP can be utilized in a plethora of messaging systems and can be provided by a plethora of transport protocols, the main focus of SOAP is remote process calls transported through HTTP. Dave Winer, Don Box, Bob Atkinson, and Mohsen Al-Ghosein designed SOAP as an object-access protocol for Microsoft in 1998. However, this new specification was not available publicly and it was submitted to the Internet Engineering Task Force (IETF) for review on 13 September 1999.

SOAP has a very rigid structure and a message must be formatted in the required format which contains Envelope, Header, Body, and an optional element Fault. SOAP message is an ordinary XML document which contains the following elements:

1. **Envelope:** SOAP envelope is a mandatory element. It is just like a regular mail envelope that holds the letters we try to deliver.
2. **Header:** This element is optional. It is used to send attributes that can be useful for processing the message either at the intermediate point or at the end-point.
3. **Body:** This is another mandatory element which contains the actual data that is being sent in XML format.
4. **Fault:** This is an optional element which is used to provide information about message processing errors.

The following code shows the general structure of SOAP and contains the above mentioned elements:

```
<?xml version = "1.0"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV = "http://www.w3.org/2003/05/soap-envelope"
    SOAP-ENV:encodingStyle = "http://www.w3.org/2003/05/soap-encoding">
    <SOAP-ENV:Header>
        ...
        ...
    </SOAP-ENV:Header>
    <SOAP-ENV:Body>
        ...
        ...
            <SOAP-ENV:Fault>
                ...
                ...
            </SOAP-ENV:Fault>
        ...
    </SOAP-ENV:Body>
</SOAP_ENV:Envelope>
```

1.9 | Communication Between Front-End and Back-End

Front-end and back-end must use a mechanism to communicate with each other. In order to get data from web services, we need to send a request from front-end and receive a response from the back-end. In the following subsections we will highlight two formats that we can use to send and receive information between front-end and back-end.

1.9.1 JavaScript Object Notation (JSON)

JavaScript Object Notation (JSON) is a form of syntax which is used to store and transfer data on the Internet. It is plain text which takes advantage of the object notation from JavaScript. A browser and server can only send and receive data which is in the textual form. With JSON, we have the data in text. Objects in JavaScript can be easily transformed into JSON, which are then used by the server. The server can then convert JSON into an object of JavaScript. This means that there is no need for parsing.

In more formal terms, it can be said that JSON is a representation of data which does not operate on the back of schema and uses ordered lists and key–value pairs to manage data. Due to its convenience, JSON receives support from all common programming languages. Today, it exists as one of the primary mediums for the exchange of data by clients and servers in the web industry. As a basic example, let us take a look at the following program:

```
var adam =
{
    "occupation": "software developer",
    "city" : "Austin, TX",
    "country": "USA"
};
```

This program generates an object which can be accessed by "adam". The curly brackets represents the "value" of our object. An object can entail multiple properties by the use of key–value pair, where commas demarcate them. In order to get the value of any property of "adam", we can write the following:

```
document.write('Adam lives in' adam.city);// In return the output is Austin, TX.
document.write('The country of residence for Adam is' adam.country);// In return the
output is USA.
```

A single variable can also store information for multiple people. To do this, square brackets are to entail more than one object. For example, for two employees, see the following program:

```
var employees = [{
    "name": "Adam",
    "city" : "Austin, TX",
    "country": "USA"
},
{
    "name": "William",
    "city" : "Fort Worth, TX",
    "country": "USA"
}]
```

1.9.2 Extensible Markup Language (XML)

Extensible Markup Language (XML) is commonly used to describe data. Its format is flexible which assists to generate various formats for information while it is utilized to exchange data on enterprise networks and public networks. XML bears some similarity to HTML as both make the use of markup. However, while HTML is used for displaying the contents of a web page, XML is used for displaying the data. The terms self-defining and self-describing are sometimes associated with XML because the data's structure is embedded in the data and therefore there are no requirements for any pre-building of the structure needed for the storage of data.

The most fundamental component of a document written in XML is known as *element.* To define elements, tags are used. All elements have an opening tag and a closing tag. The root element is the outermost element, which encompasses all the elements of an XML document. Hence, hierarchy is supported in XML. The following is a basic XML example,

```
<?xml version="1.0" standalone="yes"?>
<conversation>
    <greeting>Hello World!</greeting>
    <response>Hello Friend!</response>
</conversation>
```

1.9.3 Database

One must have a fundamental knowledge of databases. First, there are traditional relational databases where tables are known as *relations.* Each relation has a row (or record) and a column (or attribute) to store the details about the tables. MySQL, SQL Server, and Oracle are some of the most in-demand databases for storing data. These databases use a domain-specific language known as Structured Query Language (SQL) for writing statements.

Second, we have the latest NoSQL databases. NoSQL databases are unique because they do not use SQL; in fact, NoSQL stands for "NotSQL". NoSQL databases provide a greater degree of flexibility and operational speed, but they compromise on data consistency.

In this book, we will learn to use MySQL database. Also, in order to connect to this database, we will use an Object Relational Mapping (ORM) tool known as Hibernate. Let us first understand ORM with Hibernate.

1.10 | Introduction to Object Relational Mapping (ORM) with Hibernate

The connection between relational models and object models often results in complexities because of their unique approaches. While object-oriented programming (OOP) languages such as Java use an interconnected object graph to represent data, relational database management systems work with data in a tabular format. To solve this issue, ORM came into existence. ORM allows developers to access and modify objects while it saves them from thinking about the relation of objects with data sources.

By using the OOP concept of abstraction, ORM maps details of RDBMS or XML data sources with a single or multiple objects where the updated modifications in the linked interfaces are kept hidden from the programmers.

This means that all the encapsulation happens inside of ORM. Hence, ORM updates itself when there is a change in the API or the data sources instead of letting the application to do the hard work. Therefore, developers find it convenient to use newly available classes. Figure 1.5 shows how objects in memory are linked to RDBMS via ORM's mapping logic; a simple representation to understand the ORM concept.
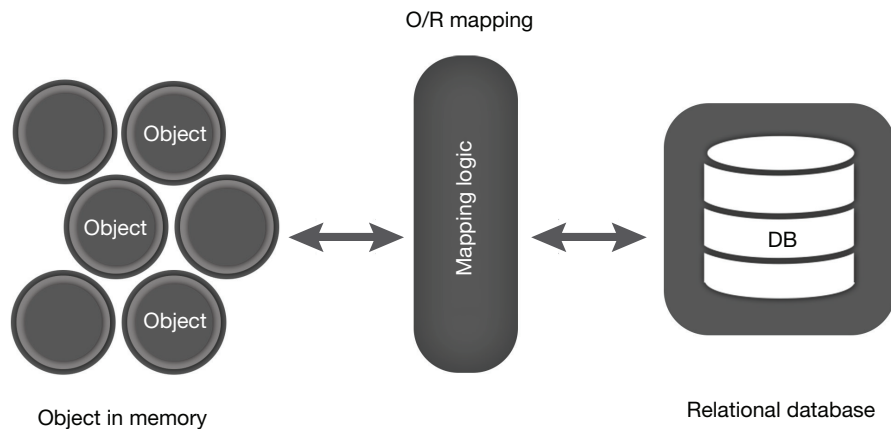

Figure 1.5   ORM mapping.

1.10.1 Hibernate

Hibernate is one of the most popular tools for ORM in Java's ecosystem. It facilitates applications to gain persistence. Persistence refers to the phenomenon in which the data in applications lasts longer than the processes of the application. This is useful when programmers desire to extend the life of any object from out of JVM's scope so they can reuse it later. As an ORM tool, its converts POJOs to tables of relational databases.

Summary

As you must have understood, in order to develop a meaningful application, one should have knowledge of the complete development stack. Since the technology is continuously evolving at a faster rate, there is a huge demand for full stack developers who cannot only understand the overall development challenges but also help in identifying the right technologies for the required purpose.

The aim of this book is to give you a practical path to learn full stack development. This will give you hands-on experience while learning the concepts. Hence, we will be taking you through a journey of developing a real-life application to see how the concepts can be applied in practice.

In this chapter, we have learned the following:

1. Meaning of full stack development.
2. Essential skills and technologies required by a full stack developer.
3. Web application development and various front-end technologies such as HTML, CSS, JQuery, and Bootstrap.
4. Back-end technologies and Java 11.
5. MVC, JSON, and XML and how to use them.
6. Web services and end-point APIs.
7. How to think of creating end-points.
8. Databases and how to use of them in the application development.
9. ORM and concept of Hibernate.

In the next chapter, we will discuss a project idea and plan out its development path. We will learn about various architectural styles and tools.

Multiple-Choice Questions

1. Which one of the following objects returns the present and updated position of the user when the user tries to make a move from one position to other?
   - **(a)** `getCurrentPosition ()`
   - **(b)** `clearWatch ()`
   - **(c)** `Timestamp`
   - **(d)** `WatchPosition()`
2. Which one of the following methods will help you capture all the click events in a window?
   - **(a)** It won't be possible to capture the events
   - **(b)** `window.routeEvents ():`
   - **(c)** `window.raiseEvents (Event.CLICK);`
   - **(d)** `window.captureEvents (Event.CLICK);`
3. _____ is also called the contaminated property of a window object in JavaScript.

- **(a)** Protocol
- **(b)** Pathname
- **(c)** Host
- **(d)** Default Status
4. A unit of a JavaScript must start with and end with _____.
   - **(a)** Semicolon, ampersand
   - **(b)** Semicolon, colon
   - **(c)** Ampersand, colon
   - **(d)** Ampersand, semicolon
5. It is possible to embed JavaScript code with HTML directly.
   - **(a)** True
   - **(b)** False

Review Questions

1. What is full stack development?
2. What are the uses of HTML and CSS?
3. How should one add Table to an HTML page and make it responsive to render properly on major devices?
4. What is the use of JavaScript?
5. Why should we use jQuery instead of plain JavaScript?
6. What is the benefit of MVC pattern?
7. What are the important elements of web development?
8. What is ORM?
9. What are the benefits of using Hibernate?
10. Why should we use web services?
11. What is REST web service?
12. What are the advantages of REST versus SOAP?
13. Why is Java better than PHP for back-end development?
14. Give five reasons why developers should use Bootstrap.

Exercises

1. Create an HTML page that shows a student report. Use tables, headings, etc.
2. Add CSS to make it look good and use different colors, font sizes, etc.
3. Add Bootstrap to make sure it renders properly on all major devices.
4. Add jQuery code to add a new record to the table.
5. Create a detailed comparison chart for JSON and XML.

Project Idea

Problem Statement: A school in a village has about 400 students in various classes. They do have Internet and electricity connections. However, the problem is their students come from far distances. Hence, they want to develop a web application which will allow them to add e-learning courses and exercises so that students do not have to come for extra classes. Design an e-Learning portal where the school can add courses and students can log in and access these courses.

Draw three-layer architecture diagram like front-end, back-end and database for the above-mentioned problem. Think of HTML pages you will need to complete the tasks and decide on database tables you will need. Also, decide on a communication technology (such as JSON or XML) you will use and explain reasons behind your selection.

Recommended Readings

1. Jennifer Preece, Helen Sharp, and Yvonne Rogers. 2015. *Interaction Design: Beyond Human-Computer Interaction, Fourth Edition*. Wiley: New Jersey

2. Mex Tegmark. 2008. *Life 3.0: Being Human in the Age of Artificial Intelligence*. Knopf: New York

3. Paul R. Daugherty. 2018. *Human + Machine: Reimagining Work in the Age of AI*. Harvard Business Review Press: Massachusetts

4. Nick Bostrom. 2016. *Superintelligence: Paths, Dangers, Strategies*. Oxford University Press: Oxford